AEC Computing and Applied Mathematics Center

AEC RESEARCH AND DEVELOPMENT REPORT

TID-4500
15th Ed.

NYO-9487
PHYSICS

SOME PROBLEMS IN LINEAR GRAPH THEORY THAT ARISE IN THE ANALYSIS OF THE SEQUENCING OF JOBS THROUGH MACHINES.

by
Jack Heller

October 15, 1960

Institute of Mathematical Sciences

NEW YORK UNIVERSITY
NEW YORK, NEW YORK

AEC Computing and Applied Mathematics Center
Institute of Mathematical Sciences
New York University

TID-4500
15th Ed.

NYO-9487
PHYSICS

SOME PROBLEMS IN LINEAR GRAPH THEORY
THAT ARISE IN THE ANALYSIS OF THE
SEQUENCING OF JOBS THROUGH MACHINES.

by

Jack Heller

October 15, 1960

NYO-9487
PHYSICS AND
MATHEMATICS

ABSTRACT

The problems of sequencing jobs through machines are discussed in a linear graph framework. The construction of feasible schedules from given technological orderings is related to the construction of transitive graphs from given component graphs. Methods of constructing transitive graphs are given and bounds on the number of different transitive graphs constructed from given components are determined. A recursive convex function defined on the transitive graphs - the job operation completion time and schedule time - is studied. Bounds on the number of different values that the schedule time can attain is obtained. Examples of multiprogramming, flow shop and machine shop scheduling are studied.

TABLE OF CONTENTS

SOME PROBLEMS IN LINEAR GRAPH THEORY THAT ARISE IN THE ANALYSIS OF THE SEQUENCING OF JOBS THROUGH MACHINES.

1. Introduction.

Questions revolving around the sequencing of jobs through machines have always been confusing because of the lack of a clear statement and formulation of the problem. In this paper we study problems connected with the construction of linear graphs which have their motivation in the sequencing of jobs through machines. To us the linear graph description is both natural and desirable: it is natural, for directed graphs[1] are convenient descriptions of the ordering of jobs through machines; it is desirable, for we are able to connect an applied problem to a developed branch of mathematics.

From the heuristic point of view, a job processed at a machine station is pictured as a node, and the order in which the jobs are processed is pictured as an oriented branch indicating which job can be processed after the completion of other jobs. For example, if we designate the nodes by a pair of integers, the first integer standing for machine station and the second integer standing for job, a pictorial representation of a job proceeding from machine station to machine station is given by each connected linear graph in figure 1. The graph on the extreme left pictures job 1 going from machine station 1, to machine station 2, ..., to machine

---

[1] For the terminology of linear graph theory, see Berge [2].

station 4. The graph on the extreme right pictures job 5 going from machine station 4 to machine station 3 to machine station 1. The ordering of the nodes as indicated by the branches is the so called technological ordering.

A schedule is constructed from the technological ordering graphs by connecting the nodes for each machine station in a fashion which describes the order of the jobs through the machine station. For example, in figure 2 we have a schedule graph. Machine station 2 processes job 3, then job 4 then job 2, and finally job 1.

As is well known, not every schedule graph is "feasible," i.e. in the terminology of Akers and Friedman[1] the schedule graph does not contain "loops" or in the linear graph terminology the schedule graph is <u>transitive</u>. One of the problems that we study in detail is the construction and counting of transitive schedule graphs.

A second problem in the sequencing of jobs through machines is the construction of "best" schedules. The "best" schedule is one which minimizes some defined function over the set of all feasible schedule graphs. The function we study is the schedule time: it is a function of the feasible schedule graph and the given processing times of each job at each machine station.

From the formal point of view we study the construction of transitive graphs from component graphs by connecting the components with directed branches. The resulting graph is such that the nodes of each chain containing the same m designation

is simply ordered;[2] the set of nodes containing the same j has the same ordering as the $j^{\underline{th}}$ component.

A lower bound on the number of topologically different transitive graphs with identified nodes is obtained. This bound is obtained by counting the number of graphs obtainable by constructions - using given components - which guarantee transitivity. These constructions and countings depend on a description of the coverings[2] of the nodes: we do not consider the connection[1] matrix of the graph, which, of course, is equivalent. The value of this approach becomes apparent when efficient algorithms for the construction of transitive graphs using digital computers is desired. The mimicking of the covering description requires much less detail than the connection matrix approach: compare [6] which uses the connection matrix approach to [11] which uses the covering approach.

A recursive convex function is defined on the transitive graphs. An upper bound for the number of different values that the function can attain over the set of transitive graphs constructed from a given set of components is obtained. As has been found in a special case [7], the number of different values of the function is small compared to the number of transitive graphs constructable from a sufficiently large set of given components.

2. Two label designation of nodes.

Of the two distinct problems in scheduling, one is concerned with the precedence relations of jobs on machines and the other

---

[2] The definitions of the various orderings used are those given by Birkhoff [4].

is concerned with the value of a given schedule. In this section we consider the case of nodes designated by a two-tuple of positive integers m and j and consider the construction of linear graphs, i.e. schedules described by precedence relations, from a set of given component linear graphs $\mathcal{J}_j$. In the simplest case the nodes of each graph are simply ordered; this case is the one we will consider in detail.

We designate each node by the two-tuple (mj). If we need to refer to a permutation of some of the nodes, we will subscript the m as m_ℓ and the j as j_k. If one node is connected to another node by a chain of nodes and branches, we write $(m'j) \preccurlyeq (m''j)$ using the binary relation $\preccurlyeq$ to indicate which node "comes first." If the two nodes are distinct and are on the same chain we write $(m'j) \prec (m''j)$ using the binary relation $\prec$ to indicate which node comes before the other node. If we designate the set of integers $1, 2, \ldots, M$ as $\mathbf{M}$, we can write

$$(2.1) \qquad \mathcal{J}_j = \left\{ (m_\ell j) \mid \ell \in \mathbf{M} \quad \text{and} \quad (m_\ell j) \prec (m_{\ell+1} j) \right\}.$$

In figure 1 we give a few pictorial examples of components $\mathcal{J}_j$. Within each node we give its (mj) designation and use the oriented arrow to indicate the precedence relation of the nodes. Thus we have

$\mathcal{J}_1$: $(11) \prec (21) \prec (31) \prec (41)$ $\quad m_1 = 1,\ m_2 = 2,\ m_3 = 3,\ m_4 = 4$

$\mathcal{J}_2$: $(12) \prec (22) \prec (32) \prec (42)$ $\quad m_1 = 1,\ m_2 = 2,\ m_3 = 3,\ m_4 = 4$

$\mathcal{J}_3$: $(23) \prec (13) \prec (43) \prec (33)$ $\quad m_1 = 2,\ m_2 = 1,\ m_3 = 4,\ m_4 = 3$

$\mathcal{J}_4$: $(24) \prec (14) \prec (44) \prec (34)$ $\quad m_1 = 2,\ m_2 = 1,\ m_3 = 4,\ m_4 = 3$

$\mathcal{J}_5$: $(45) \prec (35) \prec (15)$ $\quad m_1 = 4,\ m_2 = 3,\ m_3 = 1.$

From the given components $\mathcal{J}_j$ we want to construct a connected graph $S(P)$ such that the nodes with the same m are simply ordered and the resulting graph is transitive. (For the heuristic motivation behind this construction see Akers and Friedman[1], Heller[8] and Giffler[6].) Each chain containing a given m is of special interest, thus for each graph constructed as above we designate those chains with the same m as

$$(2.2) \qquad P_m \mathcal{M}_m = \left\{ (mj_k) \mid k \in \mathbf{J} \text{ and } (mj_k) \prec (mj_{k+1}) \right\}.$$

The set $\mathbf{J}$ is the set of the first J integers. The P_m indicates a permutation of the nodes with the same m. Thus if there is no permutation of j we write $\mathcal{M}_m = \left\{ (mj) \mid j \in \mathbf{J} \right.$ and $\left. (mj) \prec (mj+1) \right\}$.

In figure 2 we give an example of a transitive graph constructed as described above from the $\mathcal{J}_j$ given in figure 1. The sets $P_m \mathcal{M}_m$ are

$\mathcal{M}_1$: $(11)\prec(12)\prec(13)\prec(14)\prec(15)$ $j_1 = 1,\ j_2 = 2,\ j_3 = 3,\ j_4 = 4,\ j_5 = 5$

$P_2\mathcal{M}_2$: $(23)\prec(24)\prec(22)\prec(21)$ $j_1 = 3,\ j_2 = 4,\ j_3 = 2,\ j_4 = 1$

$P_3\mathcal{M}_3$: $(31)\prec(32)\prec(35)\prec(33)\prec(34)$ $j_1 = 1,\ j_2 = 2,\ j_3 = 5,\ j_4 = 3,\ j_5 = 4$

$P_4\mathcal{M}_4$: $(45)\prec(43)\prec(41)\prec(42)\prec(44)$ $j_1 = 5,\ j_2 = 3,\ j_3 = 1,\ j_4 = 2,\ j_5 = 4$.

Not every graph $S(P)$ is transitive. In figure 3 we have a graph $S(P)$ which is not transitive. The nodes (13), (12), (22) and (23) are not transitively related: $(13)\prec(12)\prec(22)\prec(23)\prec(13)$.

The following two theorems relate to the structure of the graphs $S(P)$:

<u>Theorem 1. The nodes of a transitive graph</u> $S(P)$ <u>are partially ordered by the binary relation</u> $\preccurlyeq$.

Proof. For all nodes we have $(mj)\preccurlyeq(mj)$ by the meaning of $\preccurlyeq$. If two nodes (mj) and (mj') are on the same chain $P_m\mathcal{M}_m$ then if $(mj)\preccurlyeq(mj')$ and $(mj')\preccurlyeq(mj)$ we have $(mj) = (mj')$ for $P_m\mathcal{M}_m$ is simply ordered. Similarly for two nodes (mj) and $(m'j)$ on the same chain $\mathcal{J}_j$, we have that $(mj)\preccurlyeq(m'j)$ and $(m'j)\preccurlyeq(mj)$ implies $(mj) = (m'j)$. Now let us consider a chain composed of strings of nodes $\{(m_1 j),\ (m_2 j),\ldots,(m_{\ell 1} j)\} \in \mathcal{J}_j$, $\{(m_{\ell 1} j),\ (m_{\ell 1} j_{k_1}),\ (m_{\ell 1} j_{k_1+1}),\ldots,(m_{\ell 1} j_{k_2})\} \in P_{m_{\ell_1}}\mathcal{M}_{m_{\ell_1}},\ldots;$ i.e. the strings of nodes are alternately in one of the sets $\mathcal{J}_j$ and $P_m\mathcal{M}_m$. Let us consider two nodes in this chain such that $(mj)\preccurlyeq(m'j')$ and $(m'j')\preccurlyeq(mj)$. If these nodes are in the same set $\mathcal{J}_j$ or $P_m\mathcal{M}_m$, they must be equal as shown above.

Since the given sets $\mathcal{J}_j$ are such that $\mathcal{J}_j \cap \mathcal{J}_{j'} = 0$ $j = j'$, because of the way in which we label the nodes, since $P_m \mathcal{M}_m \cap P_{m'} \mathcal{M}_{m'} = \emptyset$, because of the way we construct the graph S(P) and since $\mathcal{J}_j \cap P_m \mathcal{M}_m = (mj)$, because of the way in which we label the nodes and construct the graph S(P), we cannot have two nodes on a chain related reflexively unless they are in the same set $\mathcal{J}_j$ or $P_m \mathcal{M}_m$. Finally, since the graph S(P) is transitive, theorem 1 follows. It is not true that the nodes of any transitive graph are partially ordered. Consider the graph in figure 4: it is transitive as is easily checked, but the nodes are not partially ordered for $(11) \preccurlyeq (21)$ and $(21) \preccurlyeq (11)$ yet $(11) \neq (21)$. Graphs of this type arise in multiprogramming[5,9].

It is convenient to define the concept of a covering for nodes in a graph S(P) (cf. [4] and [8]). A node "(mj) is covered by (m'j')," if (i) (mj) and $(m'j') \in S(P)$, (ii) $(mj) \prec (m'j')$ and (iii) there does not exist an $(m''j'') \in S(P)$ such that $(mj) \prec (m''j'') \prec (m'j')$. We will use the arrow $\longrightarrow$ to indicate "is covered by" for in the pictorial representation of a linear graph the covering of one node by another is given by an oriented branch.

A transitive graph S(P) can be considered a lattice [cf. ref. 4 for a definition] with the introduction of two more nodes and suitable definitions of greatest lower bound (g.l.b.) and least upper bound (l.u.b.) between all pairs of nodes. We add a node $\mathbf{I}$ to the graph S(P) such that it is covered by all nodes in S(P) that do not cover any node in S(P); we add a node $\mathbf{O}$ to the graph S(P) such that it covers all nodes in S(P) that are not covered. We will call this graph L(P). From the graph in figure 2 we can

construct the graph L(P) by choosing $\mathbf{I}$ and $\mathbf{O}$ such that $\mathbf{I} \to (11)$, $\mathbf{I} \to (23)$, $\mathbf{I} \to (31)$, $\mathbf{I} \to (45)$, $(15) \to \mathbf{O}$, $(21) \to \mathbf{O}$ and $(34) \to \mathbf{O}$.

The g.l.b. between any two nodes (mj) and $(m'j')$ in L(P) is defined as

$$
\begin{aligned}
(2.1)\quad (mj) \frown (m'j') &= (mj) \quad \text{if} \quad (mj) \preccurlyeq (m'j') \\
&= (m'j') \quad \text{if} \quad (m'j') \preccurlyeq (mj) \\
&= (m''j'') \quad \text{if} \quad (mj) \text{ and } (m'j') \text{ are on} \\
&\quad \text{different chains, } (m''j'') \to (m_{11}j_{11}) \\
&\quad \to \cdots \to (m_{1\ell}j_{1k}) \to (mj), \\
&\quad (m''j'') \to \cdots \to (m_{2\ell'}j_{2k'}) \to (m'j') \\
&\quad \text{and } (m''j'') \text{ is the only node common} \\
&\quad \text{to the } \underline{\text{above}} \text{ chains.}
\end{aligned}
$$

The l.u.b. between any two nodes (mj) and $(m'j')$ in L(P) is defined as

$$
\begin{aligned}
(2.2)\quad (mj) \smile (m'j') &= (m'j') \quad \text{if} \quad (mj) \preccurlyeq (m'j') \\
&= (mj) \quad \text{if} \quad (m'j') \preccurlyeq (mj) \\
&= (m''j'') \quad \text{if} \quad (mj) \text{ and } (m'j') \text{ are on different} \\
&\quad \text{chains, } (mj) \to (m_{11}j_{11}) \to \cdots \to (m_{1\ell}j_{1k}) \\
&\quad \to (m''j''),\ (m'j') \to (m_{21}j_{21}) \to \cdots \to (m_{2\ell'}j_{2k'}) \\
&\quad \to (m''j'') \text{ and } (m''j'') \text{ is the only node common} \\
&\quad \text{to the } \underline{\text{above}} \text{ chains.}
\end{aligned}
$$

With these definitions we have

<u>Theorem 2. The nodes of a transitive graph</u> $L(P)$ <u>are a lattice under the binary relation</u> $\succeq$.

The proof is simple: we note $L(P)$ as defined above is partially ordered and connected. Each pair of elements in $L(P)$ has a l.u.b. and a g.l.b. in $L(P)$ which follows from the definitions of the l.u.b., g.l.b. and the fact that $L(P)$ is connected.

3. <u>Bounds on the Number of Different Transitive Graphs.</u>

One of the important properties of the graphs $S(P)$ is the number of different transitive graphs that can be constructed from the set of components $\mathcal{J}_j$. By different graphs $S(P_1)$ and $S(P_2)$ we mean that the corresponding chains in $S(P_1)$ and $S(P_2)$ with the same m have different permutations of j.

In one simple case the exact count of all graphs $S(P)$ which are transitive has been determined. In this case (cf. ref. [7] where the case was studied in detail although the counting result was not original there; cf. ref. [2]) all the sets $\mathcal{J}_j$ used in the construction of a transitive graph $S(P)$ are the same, i.e. the permutation of the m designation of the nodes is independent of j Two graphs which have the same permutation of the m designation of the nodes will be called T-O <u>equivalent.</u>[3]

We proceed to establish lemmas about the construction of transitive graphs $S(P)$. These lemmas will allow us to determine

---

[3] The notation T-O stands for technological ordering. Two jobs which have the same technological ordering have the same sequence of m designations of their nodes; i.e. the two jobs are processed on the machines in the same order.

a lower bound for the number of transitive graphs $S(P)$ that can be constructed from a set of components $\mathcal{J}_j$. The actual count seems difficult to obtain; but the lower bound obtained below will be of practical use. (For the meaning behind the desire to obtain these bounds cf. ref. [7], [10]).

Lemma 1.3. Any graph $S(P)$ is transitive if it is constructed from a T-0 equivalent set of components $\{\mathcal{J}_j | j = 1,2,\ldots,\mu\}$.

To prove this lemma we consider an arbitrary node $(m_\ell j_k) \in S(P)$. Let us consider the chains issuing from this node $(m_\ell j_k)$. These chains start as $(m_\ell j_k) \rightarrow (m_\ell j_{k+1}) \in P_{m_\ell}\mathcal{M}_{m_\ell}$ and $(m_\ell j_k) \rightarrow (m_{\ell+1} j_k) \in \mathcal{J}_{j_k}$. If we follow the chain $(m_\ell j_k) \rightarrow (m_{\ell+1} j_k) \rightarrow \cdots \rightarrow (m_m j_k) \in \mathcal{J}_{j_k}$ we can never come back to the nodes in $P_{m_\ell}\mathcal{M}_{m_\ell}$ for $\mathcal{J}_{j_k} \cap P_{m_\ell}\mathcal{M}_{m_\ell} = (m_\ell j_k)$ and the chains in $\mathcal{J}_{j_k}$ and $P_{m_\ell}\mathcal{M}_{m_\ell}$ are simply ordered. Hence if we are to arrive back at the node $(m_\ell j_k)$ by following some chain we must follow a chain which includes some nodes not in the sets $\mathcal{J}_{j_k}$ and $P_{m_\ell}\mathcal{M}_{m_\ell}$. However, once we go into a node in a set $\mathcal{J}_{j'}, (j' \neq j_k)$ the m of this node cannot be the m_ℓ of the node $(m_\ell j_k)$ because all $\mathcal{J}_j$ are T-0 equivalent; i.e. they are the same permutation on m of the nodes. Thus any two nodes $(m'j')$ and $(m''j'')$ related as $(m'j') \preccurlyeq (m''j'')$ on a chain issuing from $(m_\ell j_k)$ are related as $(m_\ell j_k) \preccurlyeq (m'j')$ and $(m_\ell j_k) \preccurlyeq (m''j'')$, which proves the lemma.

It is an easy calculation to determine the number of different transitive graphs $S(P)$ constructed from μ T-0 equivalent components $\{\mathcal{J}_j\}$. First we consider the chain with all nodes having $m = 1$. Any simple ordering of these nodes will be possible, i. e. any $P_1\mathcal{M}_1$, in some transitive $S(P)$ as given by lemma 1.3.

Hence we can have $\mu!$ different $P_1\mathscr{M}_1$ because there are μ nodes in the chain with $m = 1$. Similarly for $m = 2,3,\ldots,M$. Thus there are $\mu!^M$ different transitive graphs $S(P)$ constructable from the T-O equivalent components $\{\mathscr{J}_j | j = 1,\ldots,\mu\}$.

The next lemma gives us a method of constructing a transitive graph from a transitive graph $S_1(P)$ and a component $\mathscr{J}_{\bar{j}}$, which we will call $S_1(P) \oplus \mathscr{J}_{\bar{j}}$. (The motivation for this construction is given in [6] and [8] where methods were described for the construction of consistent schedules, i.e. in our terminology transitive graphs.)

Lemma 2.3. The graph constructed from a transitive graph $S_1(P)$ and the component $\mathscr{J}_{\bar{j}}$ such that the chains with constant m are

$$\begin{array}{c}
\cdots\to (m_1 j_{k1}) \to (m_1\bar{j}) \to (m_2 j_{k_1+1}) \to\cdots \\
\vdots \\
\cdots\to (m_\ell j_{k_\ell}) \to (m_\ell\bar{j}) \to (m_\ell j_{k_\ell+1}) \to\cdots \\
\vdots \\
\cdots\to (m_M j_{k_M}) \to (m_M j) \to (m_M j_{k_M+1}) \to\cdots
\end{array}$$

and

$$0 \le k_1 \le\cdots\le k_\ell \le\cdots\le k_M \le J$$

is transitive.

(If some of the $k_\ell = 0$, we mean that the nodes of $\mathscr{J}_{\bar{j}} \prec$ the nodes of $S_1(P)$ for the corresponding m.)

To prove this lemma, we consider any node $(m_\ell j_k)$ in the graph $S_1(P) \oplus \mathscr{J}_{\bar{j}}$ and the chains issuing from $(m_\ell j_k)$. If these chains are in $S_1(P)$ we cannot come back to node $(m_\ell j_k)$ for

the nodes of $S_1(P)$ are by theorem 1 partially ordered. If we follow a chain issuing from $(m_\ell j_k)$ and arrive at a node $(m'\bar{j}) \in \mathcal{J}_{\bar{j}}$ and follow the chain in $\mathcal{J}_{\bar{j}}$ we may come to a node $(m_\ell j'') \in P_{m_\ell}\mathcal{M}_{m_\ell}$ for $\mathcal{J}_{\bar{j}} \cap P_{m_\ell}\mathcal{M}_{m_\ell} = (m_\ell \bar{j})$. This chain is of the form $(m_\ell j_k) \rightarrow \cdots \rightarrow (m_{\ell_1} j_k) \rightarrow \cdots \rightarrow (m_{\ell_1} j_{k_{\ell_1}}) \rightarrow (m_{\ell_1}\bar{j}) \rightarrow (m_{\ell_1+1}\bar{j}) \cdots \rightarrow (m_\ell \bar{j})$. From the conditions of the lemma $(m_\ell j_k) \prec (m_\ell \bar{j})$ for we have followed chains in which the appearance of nodes in $\mathcal{J}_{\bar{j}}$ are simply ordered and each node in its order in $\mathcal{J}_{\bar{j}}$ appears further along in its corresponding $P_m\mathcal{M}_m$. Thus if $(m'j')$ is any node on a chain issuing from the node $(m_\ell j_k)$, we have $(m_\ell j_k) \prec (m'j')$ and if $(m'j') \prec (m_\ell \bar{j})$ then $(m_\ell j_k) \prec (m_\ell \bar{j})$, which was to be proven.

By this construction, we may construct many transitive graphs. For example, in figure 5 we have a transitive graph $S(P_1)$ and a component $\mathcal{J}_{\bar{j}}$ from which we construct 6 transitive graphs.

There are methods of obtaining different transitive graphs from a given transitive graph satisfying certain properties:

Lemma 3.3. *Let $S(P)$ be a transitive graph containing a set of nodes $\{(mj_k) \mid k = k_0, k_1 + 1, \ldots, k_f\}$ having the following properties:*

(i) the nodes $\{(mj_k)\}$ are covered by the set of nodes $\{(m'j')\}$ such that $m' < m$ and

(ii) the nodes $\{(mj_k)\}$ cover the set of nodes $\{(m''j'')\}$ such that $m < m''$. Then any graph constructed from $S(P)$ and a permutation of the nodes $\{(mj_k)\}$ leaving $P_m\mathcal{M}_m$ simply ordered will be transitive.

Let us consider a graph constructed as described in the lemma and divide the nodes into the following sets: (i) A_1, those nodes for which the first index is $< m$; (ii) A_2, those nodes for which the first index is $> m$; (iii) S_0, those nodes for which the first index is $= m$ and $(mj') \prec (mj_k) \in \{mj_k\}$; and (iv) S_f, those nodes for which the first index is $= m$ and $(mj_k)(\in \{(mj_k)\}) \prec (mj'')$. These four sets of nodes and the nodes $\{(mj_k)\}$ comprise all the nodes of the graph $S(P)$ (see figure 6). Let us consider a node $(mj_{k'})$ in the set $\{(mj_k)\}$ constructed as described in the lemma. If we follow a chain issuing from $(mj_{k'})$ whose next node has a first index $\neq m$, we find ourselves in the set of nodes A_2. There may be a chain which comes to a node whose first index $= m$. This node $(mj'') \in S_f$ must be in the set S_f because by hypothesis all nodes in $\{(mj_k)\}$ are covered by nodes in A_2 and the original graph $S(P)$ is transitive. Hence for any node $(mj_{k'}) \in \{(mj_k)\}$ is such that $(mj_{k'}) \prec (mj'')$.

Following a chain issuing from (mj'') we could enter the set of nodes A_1. However, the chains issuing from a node in S_f could not enter the set of nodes $\{(mj_k)\}$ since the original graph $S(P)$ was transitive. Thus the transitive relation holds for every node issuing from a node in $\{(mj_k)\}$ to the set of nodes in A_2 to the set of nodes S_f and finally to the set of nodes A_1. If a chain skips the set A_2 the above argument still holds.

It remains to show that there is no chain which goes from a node in $\{(mj_k)\}$ to a node in S_0. However, this possibility

is ruled out, since the original graph $S(P)$ was transitive and the set of nodes $P_m \mathcal{M}_m$ is simply ordered.

With the use of these lemmas we are able to obtain a lower bound on the number of transitive graphs that can be constructed from the set of components $\{\mathcal{J}_j\}$.

Theorem 3. Let the set of components $\{\mathcal{J}_j | j = 1,2,\ldots,J\}$ be divided into T-O equivalent sets $\{\mathcal{J}_j | j_1 = 1,2,\ldots,\mu_1\}$, $\{\mathcal{J}_{j_2} | j_2 = \mu_1 + 1, \mu_1 + 2,\ldots \mu_1 + \mu_2\} \cdots \{\mathcal{J}_{j_\tau} | j_\tau = \mu_1 + \cdots + \mu_{\tau-1} + 1, \ldots \mu_1 + \cdots + \mu_\tau = J\}$. A lower bound to the number of different transitive graphs[4] $N(S)$ that can be constructed from the set of components $\{\mathcal{J}_j\}$ such that the chains with constant m are simply ordered is given by

$$(3.1) \quad (\mu_1!\mu_2!\cdots\mu_\tau!)^M \prod_{k=\mu_1+1}^{\mu_2+\mu_3+\cdots+\mu_\tau} \left(\sum_{\ell_{m_1 j_k}=0}^{\ell_{m_1 j_{k-1}}} \; \sum_{\ell_{m_2 j_k}=0}^{\ell_{m_1 j_k}} \; \sum_{\ell_{m_3 j_k}=0}^{\ell_{m_2 j_k}} \cdots \sum_{\ell_{m_M j_k}=0}^{\ell_{m_{M-1} j_k}} 1 \right) \leq N(S)$$

where

$$(3.2) \qquad \ell_{m_1 j_{\mu_1+\mu_2+\cdots+\mu_p}} = \mu_1 + \mu_2 + \cdots + \mu_p \qquad p=1,2,\ldots,\tau.$$

The proof of this theorem follows from the counting of graphs constructed from transitive graphs and components $\mathcal{J}_j$ as allowed in lemma 2.3. The graph we start with is one of the transitive

[4] We use the notation $N(x)$ to mean the number of different x.

graphs, S_1 say, that can be constructed from the components $\{\mathcal{J}_{j_1} \mid j = 1,2,\ldots,\mu_1$ as allowed by lemma 3.1 and the component $\mathcal{J}_{\mu_1+1}$. The node $(m_1 j_{\mu_1+1}) \in \mathcal{J}_{\mu_1+1}$ can be placed in any position on the chain with constant $m = m_1$ in S_1 keeping the resulting $P_{m_1}\mathcal{M}_{m_1}$ simply ordered. There are $\sum_{\ell_{m_1 j_{m_1+1}}=0}^{\mu_1} 1$ different positions to place $(m_1 j_{\mu_1+1})$ and each position will give rise to a different final graph irrespective of how the remaining nodes are joined. By lemma 2.3 we can place each node of the component $\mathcal{J}_{\mu_1+1}$ in its corresponding chain, keeping the $P_m\mathcal{M}_m$ simply ordered, in

$$\sum_{\ell_{m_1 j_{\mu_1+1}}=0}^{\mu_1} \; \sum_{\ell_{m_2 j_{\mu_1+1}}=0}^{\ell_{m_1 j_{\mu_1+1}}} \; \sum_{\ell_{m_3 j_{\mu_1+1}}=0}^{\ell_{m_2 j_{\mu_1+1}}} \cdots \sum_{\ell_{m_M j_{\mu_1+1}}=0}^{\ell_{m_{M-1} j_{\mu_1+1}}} 1$$

different ways. (In figure 5 we see that this construction amounts to placing node (23) such that $(23)\prec(21)$ or $(21)\prec(23)\prec(22)$ or $(21)\prec(22)\prec(23)$. In the case of $(23)\prec(21)$ the node (13) can be placed such that $(13)\prec(11)$ or $(11)\prec(13)\prec(12)$ or $(11)\prec(12)\prec(13)$; in the case of $(21)\prec(23)\prec(22)$ the node (13) can be placed such that $(11)\prec(13)\prec(12)$ or $(11)\prec(12)\prec(13)$; and in the case of $(21)\prec(22)\prec(23)$ the node (13) can be placed such that $(11)\prec(12)\prec(13)$ only.)

Continuing on, we consider the node $(m_1 j_{\mu_1+2})$. By lemma 2.3, we can place each node of the component $\mathcal{J}_{\mu_1+2}$ in its corresponding

chain in $S_1 \oplus \mathcal{J}_{\mu_1+1}$ keeping $P_m^{\mathcal{m}}{}_m$ simply ordered, in

$$\sum_{\ell_{m_1 j_{\mu_1+2}}=0}^{\ell_{m_1 j_{\mu_1+1}}} \sum_{\ell_{m_2 j_{\mu_1+2}}=0}^{\ell_{m_1 j_{\mu_1+2}}} \sum_{\ell_{m_3 j_{\mu_1+2}}=0}^{\ell_{m_2 j_{\mu_1+2}}} \cdots \sum_{\ell_{m_M j_{\mu_1+2}}}^{\ell_{m_{M-1} j_{\mu_1+2}}} 1$$

different ways, if we want to consider $\mathcal{J}_{\mu_1+1}$ and $\mathcal{J}_{\mu_1+2}$ combined in some manner as allowed by lemma 1.3. We can continue on in this fashion until we have combined all the components in the T-O equivalent set $\{\mathcal{J}_j | j = \mu_1 + 1, \mu_1 + 2, \ldots, \mu_1 + \mu_2\}$. By lemma 1.3 we can start with $\mu_1!^M$ different graphs S_1 and again by lemma 1.3 we have $\mu_2!^M$ relatively different ways of choosing the $\{\mathcal{J}_j | j = \mu_1 + 1, \ldots, \mu_1 + \mu_2\}$ in the above construction. The product of all these expressions gives us the number of different transitive graphs that can be formed by the above construction from the two sets $\{\mathcal{J}_j | 1, 2, \ldots, \mu_1\}$ and $\{\mathcal{J}_j | j = \mu_1 + 1, \mu_1 + 2, \ldots, \mu_1 + \mu_2\}$. Continuing on in this fashion, we obtain the expression on the left of (3.1).

The inequality is necessary for lemma 3.3 gives us another way of constructing transitive graphs which is not included in the above construction. We note that the case illustrated in figure 7 gives three different graphs which is just the value we obtain from the left side of (3.1). Hence the equality can sometimes be reached and finally proves the theorem.

The interest in the number of different transitive graphs of

the above type was always an oddity stated to show that the problem of finding the minimum of a certain function defined over these graphs could not be found by enumeration even with large scale digital computers (cf.ref.[2]). However, it was never realized that the function of interest took on relatively few different values as compared to $N(S)$, until the numerical experiments described in [6,8] were analysed. We proceed to state and study the function of interest in scheduling defined on the transitive graphs S.

4. A Recursive Function Defined on a Transitive Graph $S(P)$.

A function of interest in scheduling theory is the time to finish processing the j_k^{th} job on the m^{th} machine. In our framework this function $T(mj_k)$ is defined at each node of each transitive graph. All other functions defined for schedules, i.e. transitive graphs $S(P)$, that are of interest are functions of this basic function $T(mj_k)$ [cf.refs. 2,6,7,8]. We will study one of these functions: the schedule time.

The function $T(mj_k)$ is defined as

$$(4.1) \qquad T(mj_k) = \max_{\substack{m' \ni \\ (m'j') \to (mj_k)}} T(m'j') + t_{mj_k}$$

where each quantity $t_{mj_k} \geq 0$ is attached to the node (mj_k) and is assumed independent of the position of the node in each transitive graph $S(P)$.[5] The function $T(mj_k)$ is easily computed

[5] The quantity t_{mj_k} is the processing time of the j_k^{th} job on the m^{th} machine.

for a given graph $S(P)$ for it is defined recursively on the chains of the graph and by our construction of $S(P)$ from the components $\{\mathcal{J}_j\}$, $S(P)$ is connected. We need only know the first term on the right hand side of (4.1) when the nodes $(m'j')$ are not covered in order to evaluate all $T(mj_k)$. We take[6]

$$(4.2) \qquad \max_{\substack{m' \ni \\ (m'j') \rightarrow (mj_k)}} T(m'j') = 0 \quad \text{if} \quad \not\exists \text{ an } (m'j') \rightarrow (mj_k).$$

This case arises only if $k = 1$ for if $k > 1$ we have $(mj_{k-1}) \rightarrow (mj_k)$; however it is not true that all (mj_1) do not cover some node as can be seen in figure 2 where $(21) \rightarrow (31) \equiv (3j_1)$.

The function of $T(mj_k)$ that we will study is defined as

$$(4.3) \qquad \mathcal{T}(S) = \max_{m'} T(m'j_J),$$

which is usually called the schedule time for the schedule $S(P)$. The originally stated problems in scheduling theory required the finding of a transitive graph $S(P)$ such that $\mathcal{T}(S)$ is a minimum over the set of all transitive graphs that can be constructed from the given components $\{\mathcal{J}_j\}$ and node values t_{mj}. In only one very special case[11] has a practical algorithm been given for the finding of such a transitive graph $S(P)$. Other methods based on integer linear programming [13,15] have been suggested but as yet have been computationally too large.

However, simulation - in our context, the evaluation of $\mathcal{T}(\mathcal{S})$

---

[6] In some investigations the right hand side of (4.2) is taken as $h_m \geq 0$ [14]. Since this adds nothing conceptually new, we take $h_m = 0$.

over a small set of the transitive graphs S(P), - gives an idea of the distribution of $\mathcal{T}(S)$ over the values of $\mathcal{T}(S)$ and can give us a method of finding a probable minimum $\mathcal{T}(S)$. The reason for this fortuitous happening is that there are very many more different transitive graphs S(P) constructed from a given set of components $\{\mathcal{J}_j\}$ than there are different values $\mathcal{T}(S)$. We proceed to study the function $\mathcal{T}(S)$ and establish these pertinent results.

For a given set of components $\{\mathcal{J}_j\}$ we have a lower limit to the number of different transitive graphs S(P) that can be constructed (theorem 3). The following theorem gives an upper limit to the number of different values that $\mathcal{T}(S)$ can attain. This result is purely combinatorial for it depends only on M the number of nodes in a component $\mathcal{J}_j$ and J the number of components. The result was originally proven[6] for a set of T-O equivalent components $\{\mathcal{J}_j\}$, but as we will see the result is true for an arbitrary set of components $\{\mathcal{J}_j\}$.

Theorem 4. The number of different values that $\mathcal{T}(S)$, can attain, as a function of transitive graphs S constructed from a set of components $\{\mathcal{J}_j\}$ is given by the inequality

$$N(\mathcal{T}) \leq \sum_{n=J}^{M\cdot J} \binom{M\cdot J}{n} \quad . \tag{4.4}$$

The proof consists in counting the possible number of different sums that go into the evaluation of $\mathcal{T}(S)$. By (4.1), we see that each $T(mj_k)$ is the sum of a subset of the $\{t_{mj}\}$. Hence, by the

definition of $\mathcal{T}(S)$ (4.3), $\mathcal{T}(S)$ is a sum of a subset of the $\{t_{mj}\}$. The least number of terms that can be in $\mathcal{T}(S)$ is J as can be seen by the following argument: If (mj_1) is not covered by any node and $\max\limits_{\substack{m' \ni \\ (m'j') \to (mj_2)}} T(m'j') = T(mj_1) = t_{mj_1}$, we have $T(mj_2) = t_{mj_1} + t_{mj_2}$. Continuing on in this manner, we find that we could have $T(mj_J) = t_{mj_1} + t_{mj_2} + \cdots + t_{mj_J}$. This sum is a possible candidate for the value $\mathcal{T}(S)$ and indeed may be the value. The largest number of terms possible for some transitive graph S is $M \cdot J$, the total number of nodes.

We now consider the number of sums of n numbers t_{mj} that can be chosen from $M \cdot J$ numbers t_{mj}. If all the t_{mj} were of different order of magnitude, each of these sums would be different. The total number of such sums is $\binom{M \cdot J}{n}$ and since n ranges from J to MJ, we obtain the result (4.4) of the theorem.

The number of different transitive graphs that can be constructed from a set of T-O equivalent components $\{\mathcal{J}_j\}$ is $J!^M$ (lemma 1.3). The number of different values that $\mathcal{T}(S)$ can attain as a function of these transitive graphs S is $< 2^{MJ} = \sum\limits_{n=0}^{MJ} \binom{MJ}{n}$. For moderate M and J, $2^{MJ} \ll J!^M$.

We now proceed to find estimates which depend on $\{t_{mj}\}$. These estimates are of greater practical value than the estimate (4.4), since the t_{mj} given in applications differ but moderately not by orders of magnitude. If the t_{mj} were all integral, then $\mathcal{T}(S)$ would be integral. We base our next estimate on this fact. In applications if the t_{mj} are rational, we can always multiply through by an integer so that all the t_{mj} are integers.

First we prove a theorem which gives a lower limit to the schedule time $\mathcal{T}(S)$. In practice we want one S which gives the minimum $\mathcal{T}(S)$ and so the result is of importance in its own right.

<u>Theorem 5. A lower limit to $\mathcal{T}(S)$ evaluated over all the transitive graphs constructed from the set of components $\{\mathcal{J}_j\}$ having the set of numbers $\{t_{mj}\}$ attached to its nodes is given by</u>

$$(4.5)\qquad \max_{m_\ell} \sum_{k'+1}^{J} t_{m_\ell j_{k'}} + \min_{j_1} \sum_{m_{\ell'}<m_\ell} t_{m_{\ell'} j_1} + \min_{j_J} \sum_{m_\ell<m_{\ell'}} t_{m_{\ell'} j_J} \le \min_S \mathcal{T}(S)$$

<u>where the m_ℓ in the second and third sum is that m_ℓ which maximizes the first sum.</u>

We note that T(S) is a sum of t_{mj} chosen by following each chain of S starting from a node which does not cover any nodes and ending in a node which is not covered by any node. In fact by (4.3) it is the maximum such chain. Formally

$$(4.6)\qquad \mathcal{T}(S) = \operatorname*{Max}_{\mathcal{C}} \sum_{(m'j')\in\mathcal{C}} t_{m'j'}$$

where each chain $\mathcal{C}$ is such that $(m'j')\rightarrow(m''j'')\rightarrow\cdots\rightarrow(m'''j''')$ and $(m'j')$ does not cover any node in S and $(m'''j''')$ is not covered by any node in S.

We want an underestimate to

$$\min_S \mathcal{T}(S) = \min_S \max_{\mathcal{C}} \sum_{(m'j')\in\mathcal{C}} t_{m'j'} \tag{4.7}$$

Since we are looking for the $\min\limits_S(\cdot)$, any underestimate of $(\cdot)$ will give an underestimate to $\min\limits_S \mathcal{T}(S)$. One possible underestimate is obtained by considering a chain of the form $(m_1 j)\to(m_2 j) \to\cdots\to(m_\ell j_1)\to(m_\ell j_2)\to\cdots\to(m_\ell j_J)\to(m_{\ell+1} j_J)\to\cdots\to(m_M j_J)$. Since each transitive graph contains a chain of this form,

$$\max_{m_\ell}\sum_{k'=1}^{J} t_{m_\ell j_{k'}} + \sum_{m_{\ell'}<m_\ell} t_{m_{\ell'} j'} + \sum_{m_\ell<m_{\ell'}} t_{m_{\ell'} j_J} \le \max\sum_{(m'j')\in\mathcal{C}} t_{m'j'}. \tag{4.8}$$

The first term is independent of all transitive graphs but the second and third terms depend on the choice of the transitive graph S. This dependence is reflected in the j_1 of the second term and j_J of the third term. Taking the minimum of both sides of (4.8) gives us the estimate of theorem 5.

There are, no doubt, other possible estimates, but this estimate seems adequate as can be attested to by the example given in [6,8]. In this example the equality in (4.5) was found to hold and a transitive graph giving the minimum $\mathcal{T}(S)$ was found by random sampling from the set of transitive graphs.

We can now give an estimate to the number of different $\mathcal{T}(S)$:

Theorem 6. *If the t_{mj} are integral, the number of different values that $\mathcal{T}(S)$ can attain as a function of the transitive*

graphs S constructed from the set of components $\{\mathcal{J}_j\}$ satisfies the inequality

$$(4.9)\quad N(\mathcal{T}) \leq \sum_{m'+1}^{M} \sum_{j'=1}^{J} t_{m'j'} - \max_{m_\ell} \sum_{j'=1}^{J} t_{m_\ell j'} - \min_{j_1} \sum_{m_{\ell'} < m_\ell} t_{m_{\ell'} j_1} - \min_{j_J} \sum_{m_\ell < m_{\ell'}} t_{m_{\ell'} j_J} + 1.$$

We note that if the t_{mj} are integers $\mathcal{T}(S)$ is an integer. The number of integers between and including the minimum and maximum $\mathcal{T}(S)$ is one more that $\max_S \mathcal{T}(S) - \min_S \mathcal{T}(S)$. If we overestimate the maximum $\mathcal{T}(S)$ and underestimate the minimum $\mathcal{T}(S)$ we will overestimate their difference.

Since $\mathcal{T}(S)$ is the maximum of a sum of t_{mj}, the t_{mj} taken from a chain of a transitive graph S, the largest value that $\mathcal{T}(S)$ can attain is given by the double sum in (4.9). The next three terms comprise an underestimate of $\mathcal{T}(S)$ as given by theorem 5, which establishes the theorem.

5. Partially Ordered Components $\mathcal{J}_j$.

The first examples of scheduling studies were those in which the components $\mathcal{J}_j$ were simply ordered. These examples were studied because of their conceptual simplicity; these examples are found in practice, but the more complex case - partially ordered $\mathcal{J}_j$ - are also found in practice. We will outline the description of the nodes and the construction of the transitive graphs, leaving the detailed proofs of results similar to those of sections 2 - 4 to the interested reader. All the proofs of

the previous sections have their counterpart when the components $\mathcal{J}_j$ are partially ordered: we need only change the words "simple ordering of each component $\mathcal{J}_j$" and take into account the added notation of the node designation in the partially ordered case.

In the partially ordered case the nodes are labelled with a three-tuple of indices m, j, i. The m and j have the same meaning as before, while the i indicates[7] the $i^{\underline{th}}$ node on any chain with constant m and j. In the previous examples $i = 1$. Examples of partially ordered components $\mathcal{J}_j$ are given in figure 8.

As in the case of the simply ordered components, we are interested in constructing transitive graphs from the partially ordered components $\mathcal{J}_j$ such that the nodes with the same m are simply ordered. When we succeed in constructing such a graph S, the nodes of the graph are partially ordered (cf. theorem 1). If we add the node $\mathbf{I}$ which is covered by all nodes which do not cover nodes in S(P), a node $\mathbf{O}$ which covers all nodes in S(P) which are not covered by nodes in S(P) and define the l.u.b and g.l.b. as in section 2 (2.1,2.2), the resulting set of nodes and relations defined by $\frown$ and $\smile$ form a lattice (cf. theorem 2).

In figure 9 we give an example of a transitive graph constructed as described above from the components $\mathcal{J}_j$ given in figure 8. The corresponding $P_m \mathcal{M}_m$ are as follows

---

[7] As before m stands for machine station, j for job and i stands for the $i^{\underline{th}}$ time that job j has returned to machine station m.

$P_1\mathcal{M}_1$: $(131) \prec (121) \prec (111) \prec (122) \prec (112)$

$P_2\mathcal{M}_2$: $(221) \prec (231) \prec (211) \prec (212) \prec (222) \prec (232) \prec (213)$

$P_3\mathcal{M}_3$: $(311) \prec (321) \prec (331)$

$P_4\mathcal{M}_4$: $(421) \prec (431)$

$P_5\mathcal{M}_5$: $(511) \prec (521) \prec (512) \prec (531)$.

If we are considering a permutation of the nodes for a given m, we will write the node designation as $(mj_k i_k)$. Thus, for example, in figure 9 the nodes for $m = 1$ would be designated as follows: $(1j_1i_1) = (131)$, $(1j_2i_2) = (121)$, $(1j_3i_3) = (111)$, $(1j_4i_4) = (122)$ and $(1j_5i_5) = (112)$.

Transitive graphs can be constructed in a manner analogous to the constructions given by lemmas 1.3 - 3.3 with the modifications in notation given above. The counting results, although more intricate, can be derived in a manner similar to that of section 3.

The function of interest defined for transitive graphs $S(P)$ is

$$(5.1) \qquad T(mj_ki_k) = \max_{\substack{m' \ni \\ (m'j'i') \to (mj_ki_k)}} T(m'j'i') + t_{mj_ki_k}.$$

The quantities t_{mji} are given non-negative numbers attached to the nodes and are independent of the permutation of the nodes used in the construction of each transitive graph $S(P)$. As before, we take

$$(5.2) \qquad \max_{\substack{m' \ni \\ (m'j'i') \to (mj_1i_1)}} T(m'j'i') = 0 \quad \text{if} \quad \nexists \text{ an } (m'j'i') \to (mj_1i_1),$$

which is the counterpart of (4.2). The function $T(mj_k i_k)$ defined by (5.1) is analogous to (4.1) and indeed reduces to (4.1) if all $i_k = 1$.

The function of $T(mj_k i_k)$ of interest is similar to (4.3):

$$(5.3) \qquad \mathcal{T}(S) = \max_{m'} T(mj_{\hat{J}(m)} i_{\hat{J}(m)})$$

where J is defined as follows: for each m and j we define η_{mj} the number of nodes in the component $\mathcal{J}_j$ with the first index m. Then we have

$$(5.4) \qquad \hat{J}(m) = \sum_{j=1}^{J} \eta_{mj}$$

where $\hat{J}$ is the number of components $\mathcal{J}_j$ that go into the construction of the transitive graph $S(P)$. If all the $i_k = 1$, $\hat{J}(m) = J$ which is the case of the treatment of the simply ordered components $\mathcal{J}_j$.

The countings of the number of different values that $\mathcal{T}(S)$ can take on is similar to those given in section 4. Again we are able to conclude that there are many more different $S(P)$ than $\mathcal{T}(S)$ and the decision theoretic approach to the finding of a probable $\min_S \mathcal{T}(S)$ is feasible [cf.10].

6. Quasi-Ordered Components $\mathcal{J}_j$.

Recently a scheduling problem has been mentioned under the heading of multiprogramming [5,9]. In these problems the component

$\mathscr{P}_j$ are quasi-ordered, i.e. it is possible for $(m'j'i') \preceq (m''j''i'')$ and $(m''j''i'') \preceq (m'j'i')$ and yet have $(m'j'i') \neq (m''j''i'')$. We again sketch out the general description leaving the details for further study.

Examples of quasi-ordered components $\mathscr{P}_j$ for the three index description of the nodes are given in figure 10. We note, for example, that $(111) \preceq (211)$ and $(211) \preceq (111)$ yet $(111) \neq (211)$. The components $\mathscr{P}_j$ are transitive even though they are not partially ordered.

We now want to construct transitive graphs $S(P)$ from the quasi-ordered components $\mathscr{P}_j$ such that each chain composed of nodes with a constant m are simply ordered. This construction can be accomplished in a variety of ways. In figure 11 one such graph is illustrated. The $P_m \mathcal{M}_m$ are as follows

$P_1\mathcal{M}_1 : (111) \prec (121) \prec (122) \prec (112) \prec (123)$

$P_2\mathcal{M}_2 : (211) \prec (221) \prec (212) \prec (222) \prec (223) \prec (234)$

$P_3\mathcal{M}_3 : (311) \prec (321) \prec (332) \prec (312)$.

In the permutation notation $(1j_1i_1) = (111)$, $(1j_2i_2) = (122)$, $(1j_3i_3) = (122)$, $(1j_4i_4) = (112)$ and $(1j_5i_5) = (123)$; and similarly for the chains with $m = 2$ and 3.

It is customary in the literature [1,8] to say that the graph $S(P)$ should not contain "loops" meaning in our terminology that the graph $S(P)$ is transitive. The use of the word "loop" is a poor one for the graph in figure 11 can be considered to

contain loops, e.g. $(111) \rightarrow (211) \rightarrow (111)$, yet is an acceptable schedule of the two jobs $\mathcal{J}_1$ and $\mathcal{J}_2$. It is only the transitive nature of the graph S(P) that is necessary for its acceptance as a "feasible schedule."

The nodes of a graph S(P) constructed from quasi-ordered components $\mathcal{J}_j$ in the above manner are easily shown to be quasi-ordered: the counterpart of theorem 1. There is no simple counterpart of theorem 2: we must redefine the l.u.b. and g.l.b. and take account of the fact that S(P) is quasi-ordered.

The construction of transitive graphs S(P) from the quasi-ordered components $\mathcal{J}_j$ can be accomplished as with the previous types of components. These different graphs can be counted and lead to a large number of possible graphs for any given set of components $\{\mathcal{J}_j\}$.

In order to define functions of interest on a given transitive graph S(P) we must add a new concept not necessary in the previous cases. We will say that a set of distinct nodes form a simultaneous equivalence class $\mathcal{E}$ if each pair of nodes in $\mathcal{E}$ are such that $(m'j'i') \preccurlyeq (m''j''i'')$, $(m''j''i'') \preccurlyeq (m'j'i')$ and $(m'j'i') \neq (m''j''i'')$. If we want to indicate the simultaneous equivalence class generated by a given node (mji) we will write $\mathcal{E}(mji)$. Thus, for example, in figure 11

$$\mathcal{E}(123) = \{(123),(223),(332)\}$$

and

$$\mathcal{E}(122) = \{(122)\}.$$

The function of interest is [cf.4.1-2 and 5.1-2]

$$(6.1) \qquad T(mj_k i_k) = \max_{\substack{m' \ni \\ (m'j'i') \to \mathcal{E}(mj_k i_k)}} T(m'j'i') + t_{mj_k i_k}$$

and

$$(6.2) \qquad \max_{\substack{m' \ni \\ (m'j'i') \to \mathcal{E}(mj_1 i_1)}} T(m'j'i') = 0 \quad \text{if} \quad \nexists \; (m'j'i') \to \mathcal{E}(mj_1 i_1).$$

In figure 11 we see that

$$T(111) = t_{111}$$

$$T(211) = t_{211}$$

$$T(311) = t_{311}$$

$$T(121) = \max \begin{Bmatrix} T(111) \\ T(211) \end{Bmatrix} + t_{121}$$

$$T(221) = \max \begin{Bmatrix} T(111) \\ T(211) \end{Bmatrix} + t_{221}$$

$$\vdots$$

In the previous cases $\mathcal{E}(mj_k i_k) = (mj_k i_k)$.

The function defined over the transitive graphs $S(P)$ of interest is

$$(6.3) \qquad \mathcal{T}(S) = \max_{m'} T(mj_{\hat{J}(m)} i_{\hat{J}(m)}),$$

where $\hat{J}(m)$ is as given by (5.4).

The counting of the number of different values that $\mathcal{J}(S)$ can attain is similar to those given in section 4. Again we are able to conclude that there are many more different $S(P)$ that $\mathcal{J}(S)$ and the decision theoretic approach to finding a probable $\min_S \mathcal{J}(S)$ is feasible [cf.10].

7. Transitive Graphs $S(P)$ Whose Sets $P_m \mathcal{M}_m$ are Partially Ordered.

In some scheduling problems there are many machines at each machine station m; the disposition of the machine with the job is part of the scheduling problem. To account for this question, we designate each node with a four-tuple of integers $(mjin)$. The first three integers have the same meaning as before while the fourth integer n designates the machine at machine station m processing job j on its $i^{\underline{th}}$ return to machine station m.

The components $\mathcal{J}_j$ are specified with regard to the coverings related to the indices m, j and i; the index n is determined only when a transitive graph $S(P)$ is constructed from the given components $\mathcal{J}_j$.

For example, if we use the components given in figure 8 and if n for $m = 1$ can take on values 1 and 2, n for $m = 2$ can take on values 1, 2 and 3, and n for $m = 3$, 4 and 5 can take on values 1 only, we construct the transitive graph given in figure 12. The construction is so performed that each chain for a given n is simply ordered and the order relations of the given components $\mathcal{J}_j$ are preserved. The sets $P_m \mathcal{M}_m$ are in general partially ordered; e.g.

$$P_1 \mathcal{m}_1 : (1111) \prec (1221)$$

$$(1212) \prec (1312) \prec (1122)$$

$$(1111) \prec (1121)$$

$$(1212) \prec (1221).$$

The last two relations are determined by the transitive relation on the chain $(1111) \rightarrow (2111) \rightarrow (1122)$ and $(1212) \rightarrow (2212) \rightarrow (1221)$ respectively.

Theorems 1 and 2 are easily proven as before.

We can also have scheduling questions of the above type when the components $\mathcal{J}_j$ are quasi-ordered.[8]

The construction and counting of different transitive graphs are similar to those of lemmas 1.3 - 3.3.

The function of interest defined on a transitive graph is

$$(7.1) \qquad T(mj_k i_k n_k) = \max_{\substack{m' \ni \\ (m'j'i'n') \rightarrow (mj_k i_k n_k)}} T(m'j'i'n') + t_{mj_k i_k n_k}$$

and

$$(7.2) \qquad \max_{\substack{m' \ni \\ (m'j'i'n') \rightarrow (mj_1 i_1 n_1)}} T(m'j'i'n') = 0 \text{ if } \nexists \, (m'j'i'n') \rightarrow (mj_1 i_1 n_1),$$

which is the counterpart of (4.1-2).

A function of interest defined over all the transitive graphs $S(P)$ for a given set of components $\{\mathcal{J}_j\}$ is

---

8 These problems arise when computer parts such as channels, processing units, ... are assigned in a multiprogrammed set of programs.

$$(7.3) \qquad \mathcal{J}(S) = \max_{m,n} T(m j_{\hat{J}(m,n)} i_{\hat{J}(m,n)} n_{\hat{J}(m,n)})$$

where $\hat{J}(m,n)$ is defined as follows: we let η_{mjn} be the number of nodes with indices m, j and n in the given graph $S(P)$. Then

$$(7.4) \qquad \hat{J}(m,n) = \sum_{j} \eta_{mjn}$$

the counterpart of (5.4).

Again we can determine that there are many more different $S(P)$ than $\mathcal{J}(S)$ and the decision theoretic approach is feasible.[10]

It now becomes apparent that the order relations of scheduling problems can be described by transitive linear graphs whose nodes are designated with an n-tuple of integers, there being various order relations given for the different indices of the n-tuples.

BIBLIOGRAPHY

[1] Akers, S. B., Jr and Friedman, J., "A non-numerical approach to production scheduling problems," J. Opers. Res. Am 3, No. 4, 429-442 (1955)

[2] Bellman, R., "Some mathematical aspects of scheduling," J. S.I.A.M., Vol. 4, pp. 168-205 (1956).

[3] Berge, C., Graph Theorie, Dunod, Paris (1959)

[4] Birkhoff, G., Lattice Theory, Am. Math. Soc. Coll. Pub. XXV, New York City (1948).

[5] Codd, E. F., "Multiprogram Scheduling" Comm. A C.M. 3, 347-350 (1960); Comm A.C.M., 7, 413-418 (1960).

[6] Giffler, B, "Mathematical solution of explosion and scheduling problems" - draft copy, IBM Business Systems Research Memorandum (May, 1959).

[7] Heller, J., "Combinatorial and statistical aspects of an MXJ scheduling problem," A.E C. Research and Development Report NYO-2540 (1959).

[8] Heller, J., "Combinatorial properties of machine shop scheduling," A.E.C. Research and Development Report NYO-2879, (1959).

[9] Heller, J., "Sequencing aspects of multiprogramming," to be published.

[10] Heller, J., "Some numerical experiments for an MXJ flow shop and its decision - theoretic aspects." J. Oper. Res. Vol. 8, No. 2 (1960) pp. 178-184.

[11] Heller, J. and Logemann, G.,"An algorithm for constructing feasible schedules and computing their schedule times." To be published.

[12] Johnson, S M., "Optimal two and three stage production schedules with setup times included," Naval Res. Log. Quart. Vol. 1, (1954) pp. 61-68.

[13] Manne, A. S., "On the job-shop scheduling problem," J. Oper. Res. Vol. 8, No. 2, pp. 219-223 (1960).

[14] Sisson, R. L., "Methods of sequencing in job shops - a review," J Oper. Res. Am., Vol. 7, No. 1, pp. 10-29 (1959).

[15] Wagner, H. M., "An integer linear-programming model for machine scheduling,"Naval Res. Log. Quart., Vol. 6, No. 2, pp. 131-140(1959).

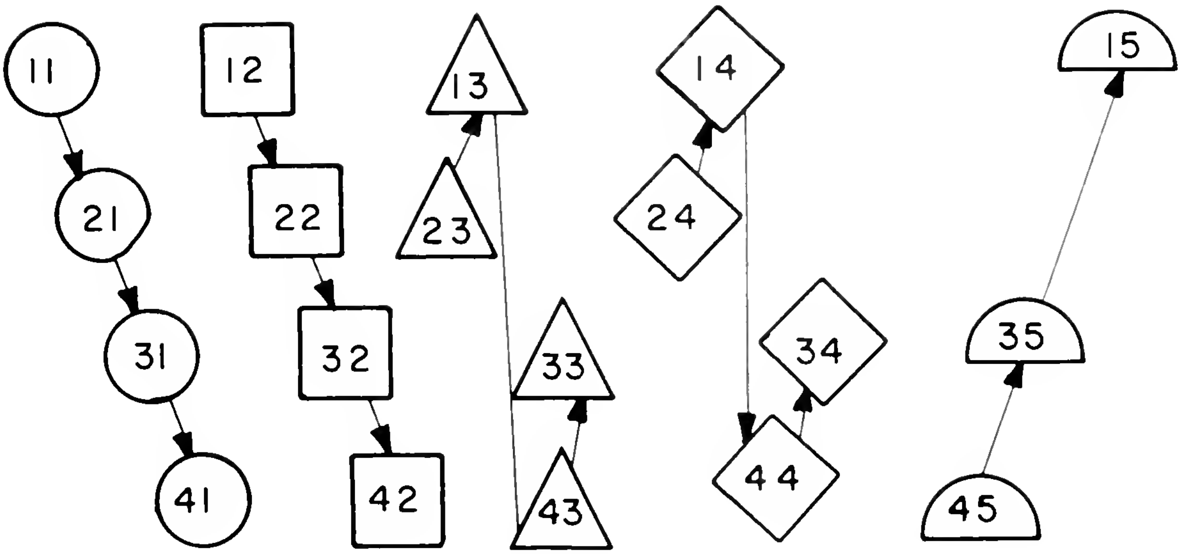


Figure 1

GRAPHS DEPICTING TECHNOLOGICAL ORDERINGS.

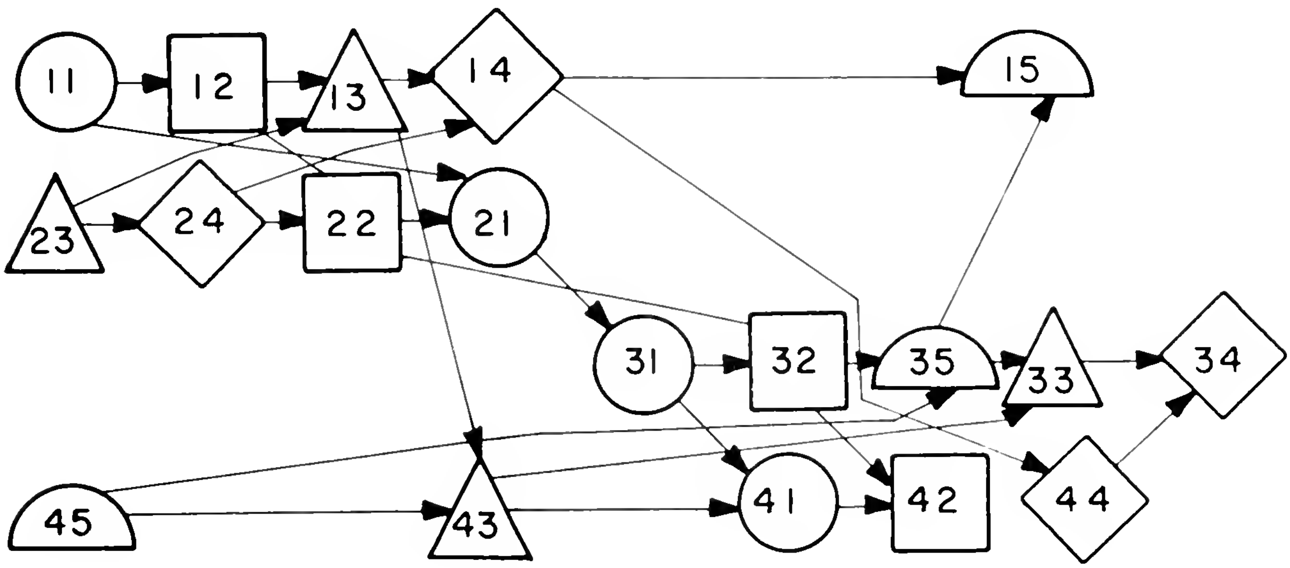


Figure 2

AN EXAMPLE OF A TRANSITIVE GRAPH CONSTRUCTED FROM THE COMPONENTS GIVEN IN FIGURE 1.

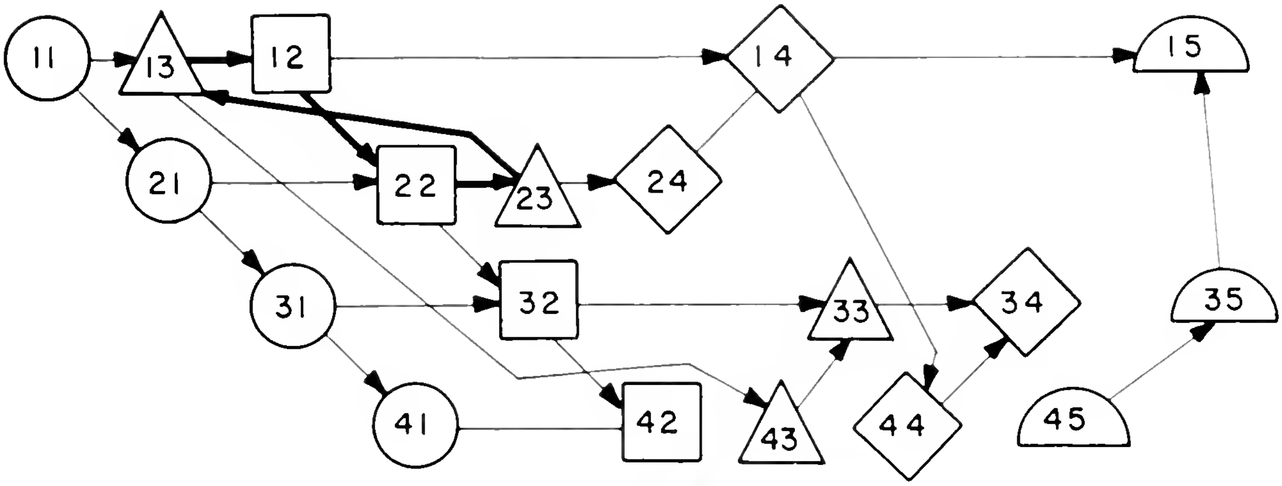


Figure 3

AN EXAMPLE OF A NON-TRANSITIVE GRAPH CONSTRUCTED FROM THE COMPONENTS GIVEN IN FIGURE I.

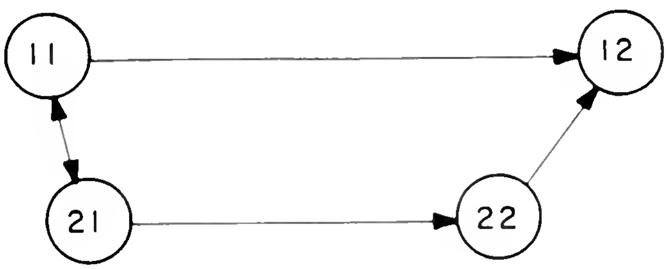
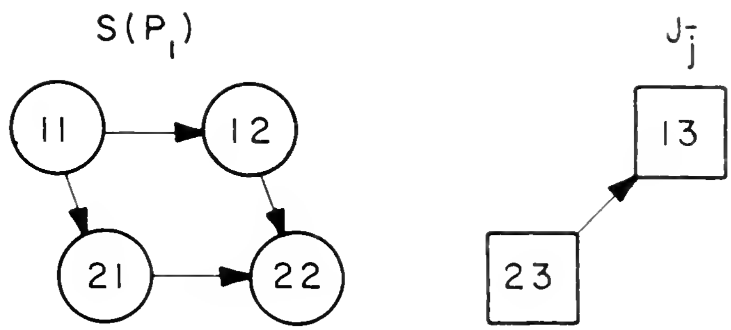


Figure 4

A TRANSITIVE GRAPH WHOSE NODES ARE NOT PARTIALLY ORDERED.



DIFFERENT TRANSITIVE GRAPHS $S(P_1) \oplus J_j^-$.

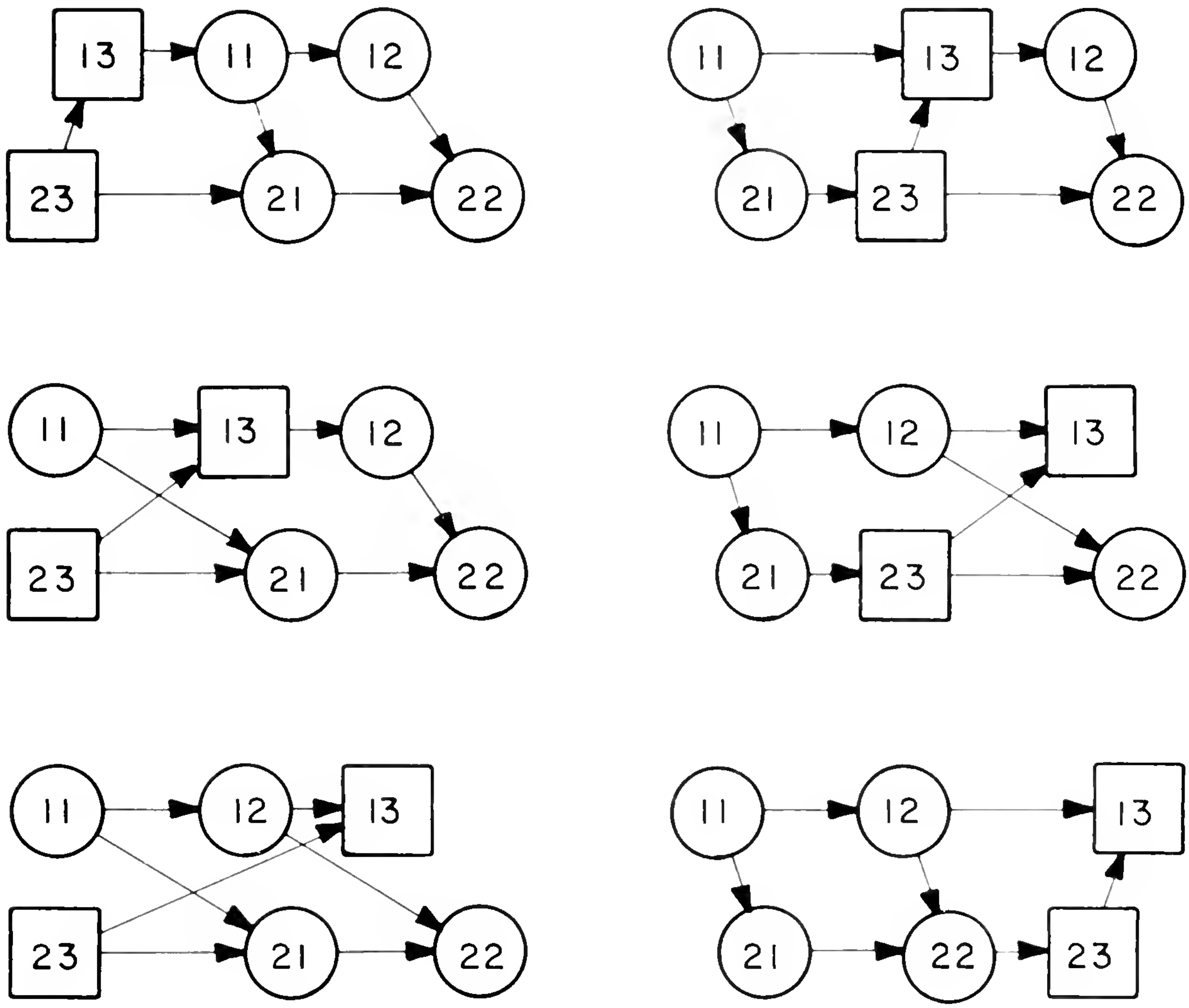


Figure 5

THE CONSTRUCTION OF DIFFERENT TRANSITIVE GRAPHS USING LEMMA 2.3.

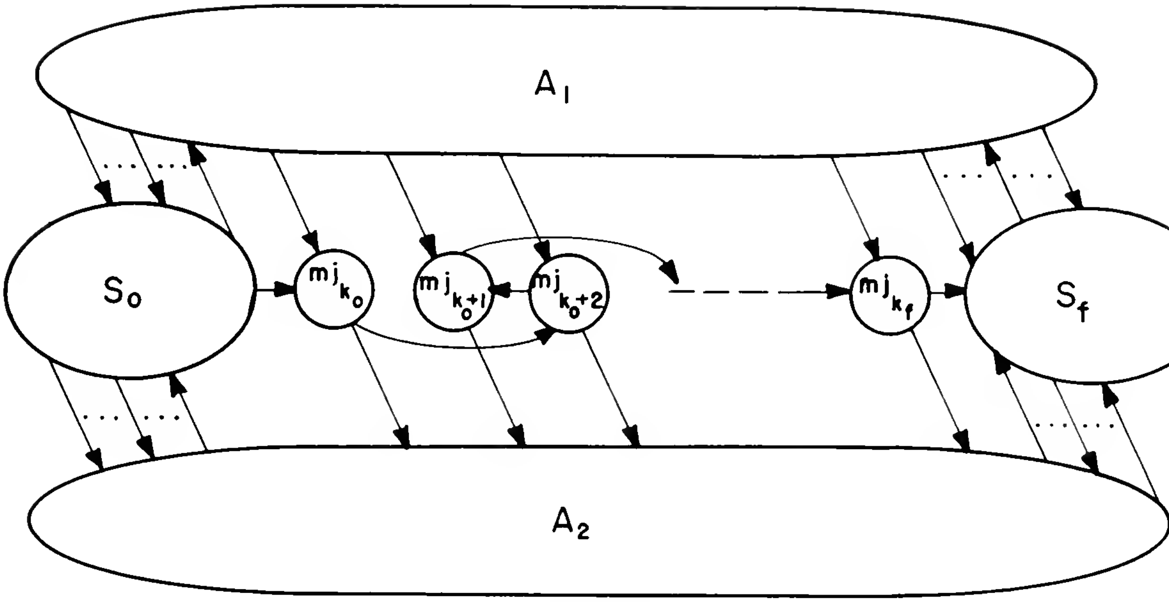


Figure 6

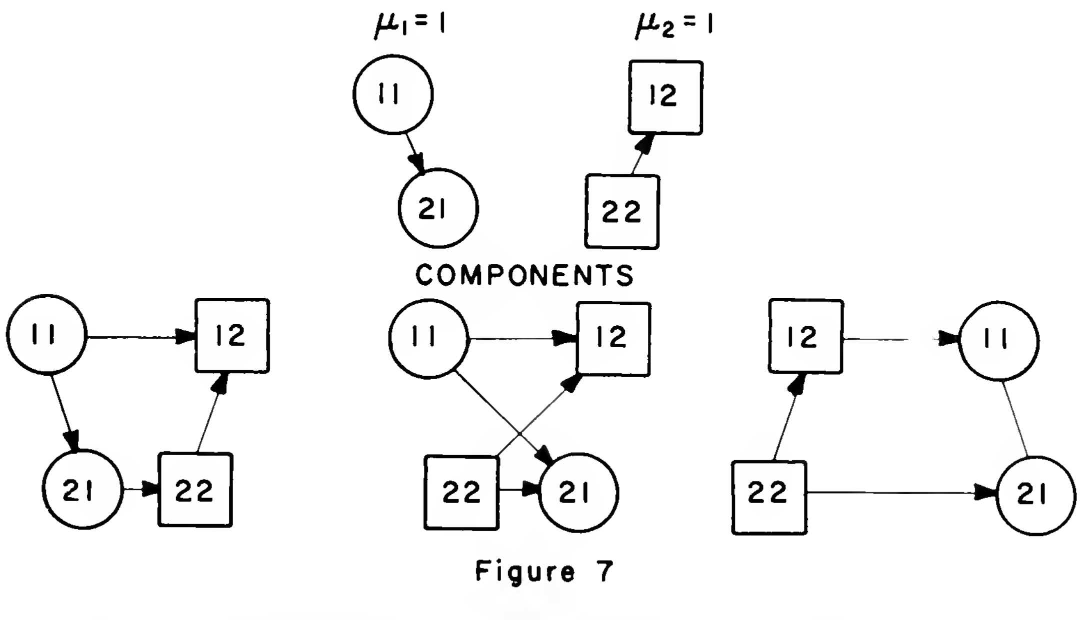


Figure 7

ALL THE DIFFERENT TRANSITIVE GRAPHS CONSTRUCTED FROM TWO COMPONENTS J_1 AND J_2.

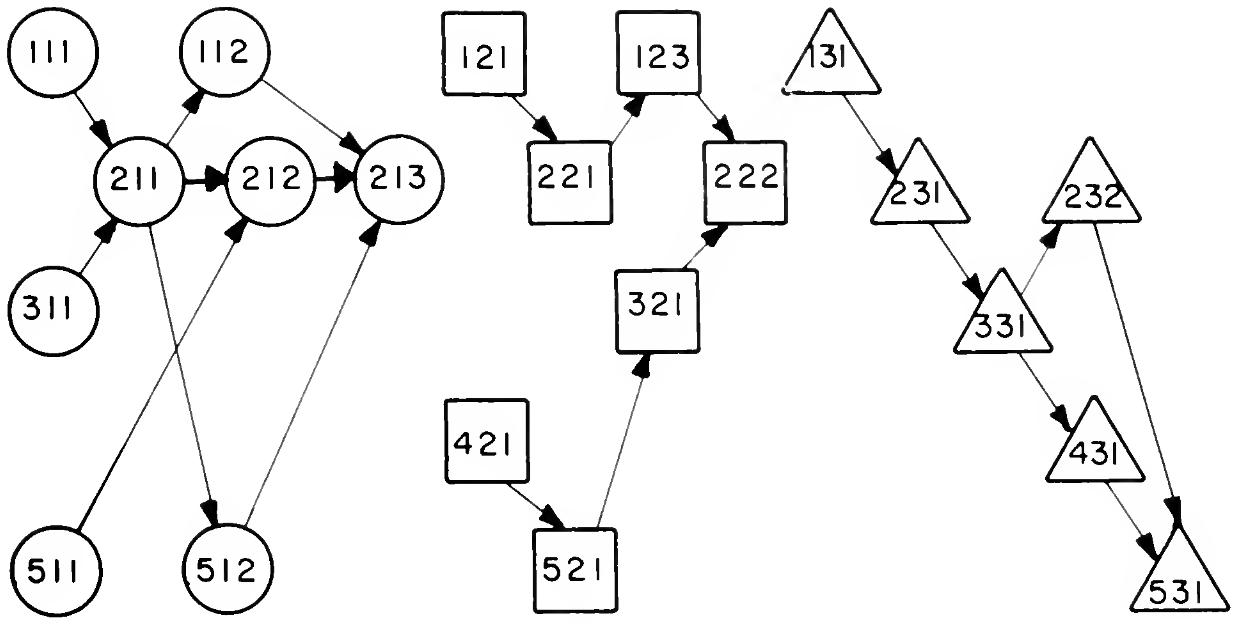


Figure 8

EXAMPLES OF PARTIALLY ORDERED COMPONENTS J_j.

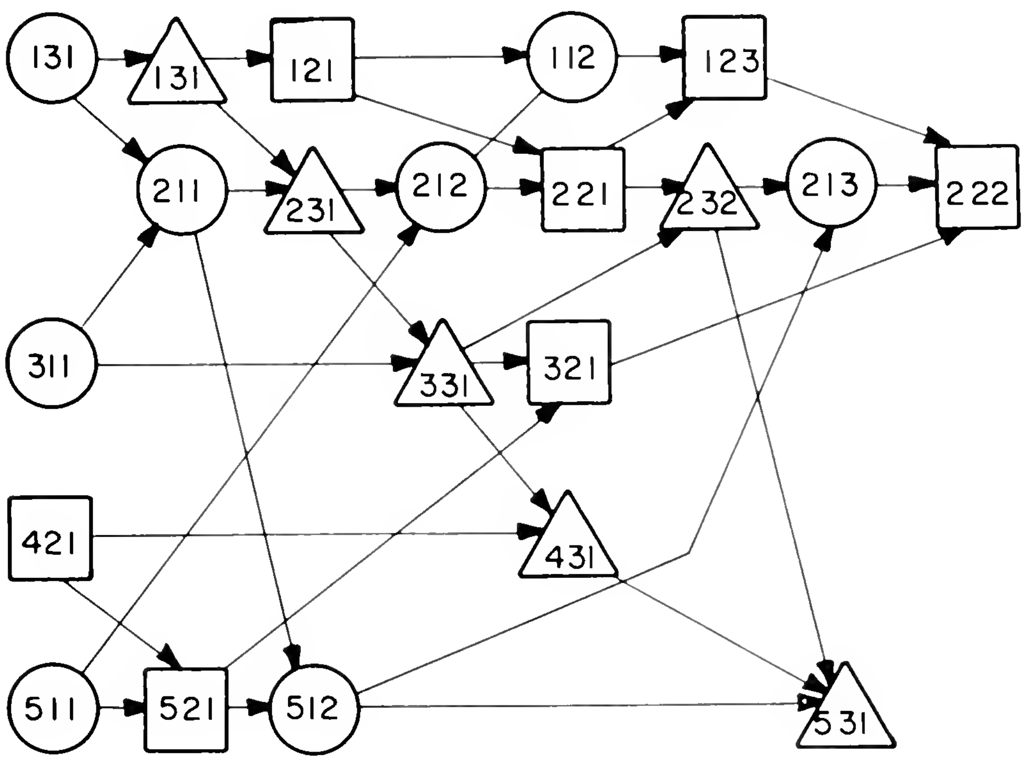


Figure 9

A TRANSITIVE GRAPH CONSTRUCTED FROM THE COMPONENTS J_j GIVEN IN FIGURE 8.

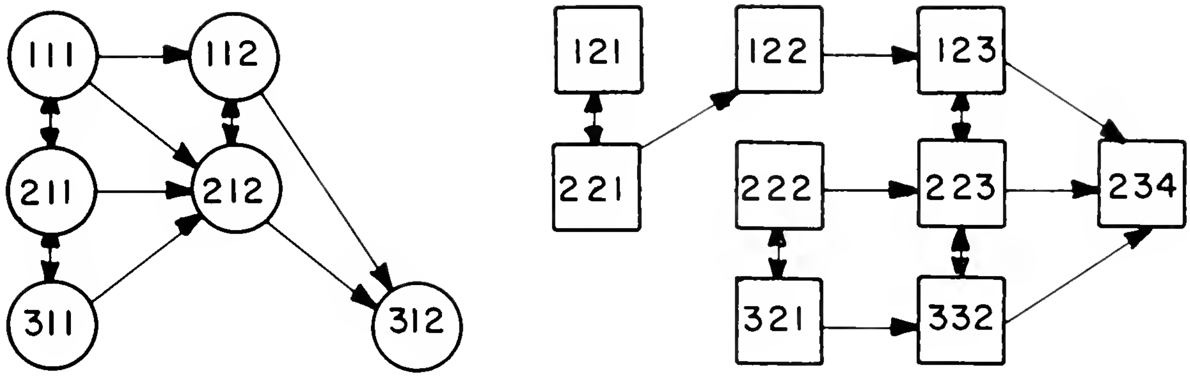


Figure 10

EXAMPLES OF QUASI-ORDERED COMPONENTS J_j.

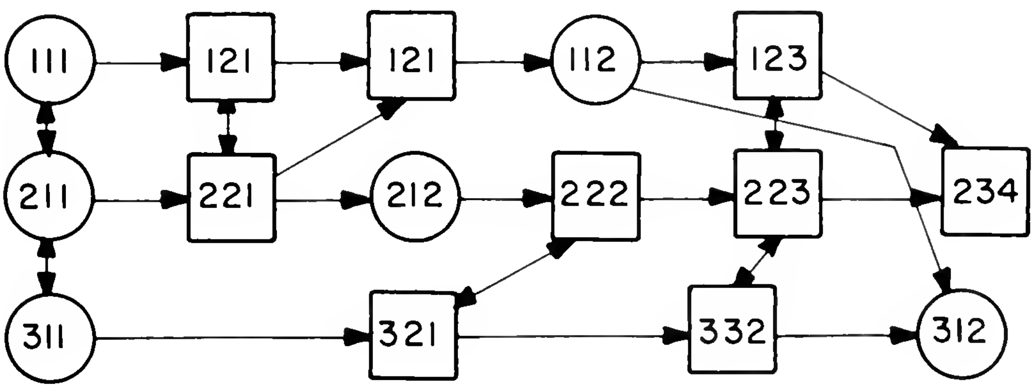


Figure 11

A TRANSITIVE GRAPH CONSTRUCTED FROM THE COMPONENTS J_j GIVEN IN FIGURE 10.

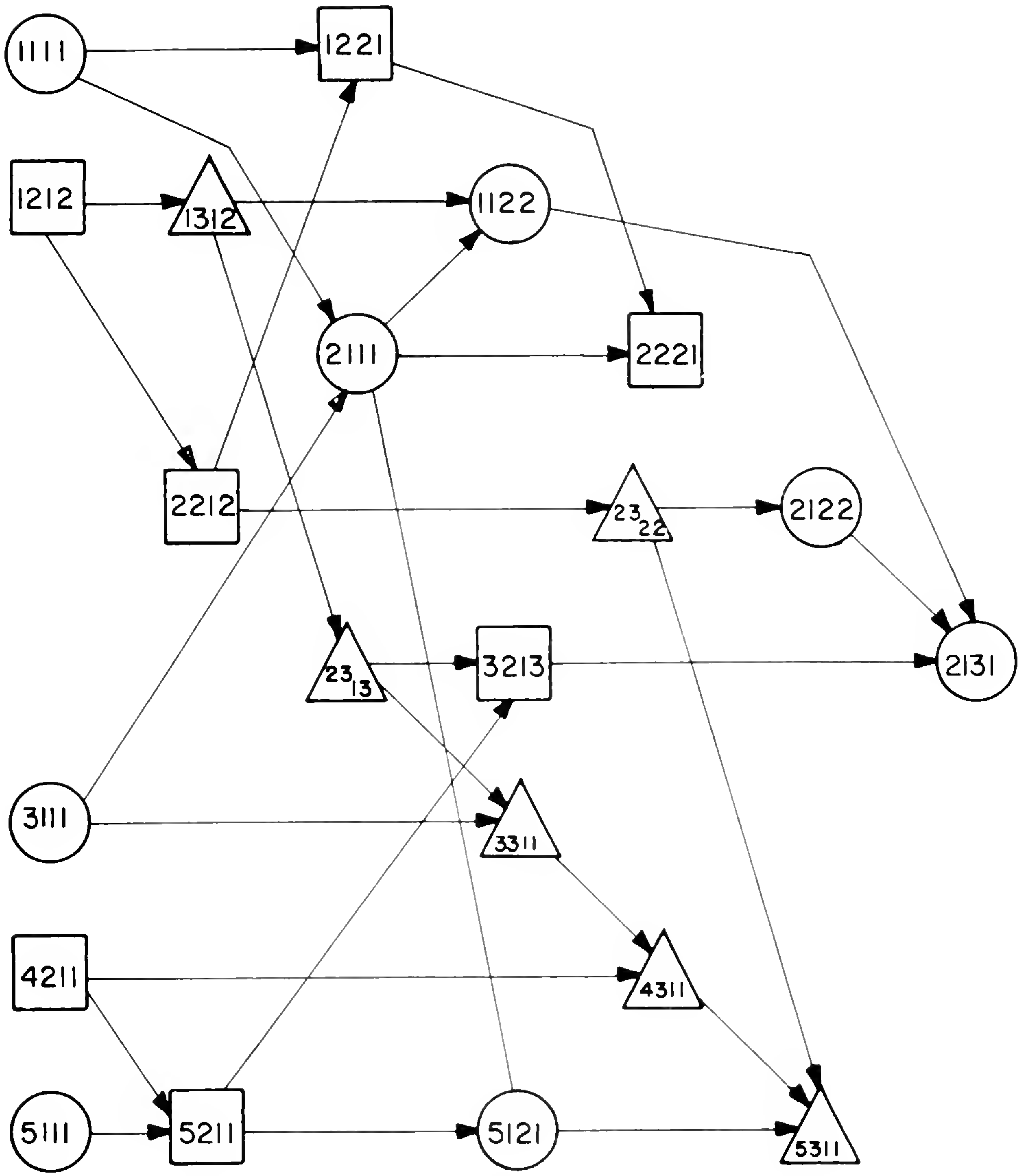


Figure 12

A TRANSITIVE GRAPH CONSTRUCTED FROM THE COMPONENTS GIVEN IN FIGURE 8, SOME OF WHOSE $P_m M_m$ ARE PARTIALLY ORDERED.